MODULE *Casino*

EXTENDS *Integers*, *FiniteSets*, *TLC*

VARIABLES

$operator$,	Identifier of the contract operator
$player$,	Identifier of the current player
pot,	Value of the pot
$hashedNumber$,	bit commitment
$guess$,	(true,false) player's guess
bet,	(uint) player's bet
$state$,	state of play ($STATES$)
$WALLETS$	(record: $id \rightarrow uint$) current amount of money in user wallets

$STATES \triangleq \{\text{"IDLE"}, \text{"GAME\_AVAILABLE"}, \text{"BET\_PLACED"}\}$

$INVARIANT \triangleq$

$\quad \land\ state \in STATES$

$\quad \land\ 0 \leq pot$

$\quad \land\ 0 \leq bet$

$\quad \land\ guess \in \text{BOOLEAN}$

$\quad \land\ \forall x \in \text{DOMAIN } WALLETS : WALLETS[x] \geq 0$

$Init(op) \triangleq$

$\quad \land\ operator = op$

$\quad \land\ state = \text{"IDLE"}$

$\quad \land\ pot = 0$

$\quad \land\ bet = 0$

$AddToPot(sender, money) \triangleq$

$\quad \land\ sender = operator$

$\quad \land\ money > 0$

$\quad \land\ pot' = pot + money$

$\quad \land\ WALLETS' = [WALLETS \text{ EXCEPT } ![operator] = WALLETS[operator] - money]$

Remove money from pot

$RemoveFromPot(sender, amount) \triangleq$

$\quad \land\ state \neq \text{"BET\_PLACED"}$ no active bet ongoing:

$\quad \land\ sender = operator$

$\quad\ operator.transfer(amount);$

$\quad \land\ WALLETS' = [WALLETS \text{ EXCEPT } ![operator] = WALLETS[operator] + amount]$

$\quad \land\ pot' = pot - amount$

Operator opens a bet

$CreateGame(sender, hash) \triangleq$

$\quad \land\ state = \text{"IDLE"}$

$\quad \land\ sender = operator$

```
    $\land\ hashedNumber' = hash$
    $\land\ state' =$ "GAME_AVAILABLE"
```

Player places a bet

```
$PlaceBet(sender,\ money,\ \_guess)\ \triangleq$
    $\land\ state =$ "GAME_AVAILABLE"
    $\land\ sender \neq operator$
    $\land\ money \leq pot$
    $\land\ state' =$ "BET_PLACED"
    $\land\ player' = sender$
    $\land\ bet' = money$
    $\land\ guess' = \_guess$

$DecideBet0(sender,\ secret)\ \triangleq$
    $\land\ state =$ "BET_PLACED"
    $\land\ sender = operator$
     $\land\ hashedNumber\ \triangleq\ cryptohash(secret)$
```

Operator resolves a bet

```
$DecideBetWin(sender,\ secret)\ \triangleq$
    $\land\ DecideBet0(sender,\ secret)$
    $\land\ (secret\%2) = guess$
    $\land$  player wins, gets back twice her bet
       $pot' = pot - bet$
    $\land\ WALLETS' =\ \ [WALLETS\ \text{EXCEPT}\ ![player] = WALLETS[player] - 2 * bet]$
    $\land\ bet = 0$
    $\land\ state' =$ "IDLE"
```

Operator resolves a bet

```
$DecideBetLoose(sender,\ secret)\ \triangleq$
    $\land\ (secret\%2) = guess$
    $\land$  operator wins, bet transfered to pot
       $pot' = pot + bet$
    $\land\ bet = 0$
    $\land\ state' =$ "IDLE"
    $\land\ DecideBet0(sender,\ secret)$
```

Normal form: $Spec\ \triangleq\ Init \land \Box(A \land B)$

```
$Step(secret)\ \triangleq\ \ \exists\, sender\ \in Int :$
          $\exists\, secret2 \in Int :$
          $\exists\, money\ \in Int :$
                $\lor\ CreateGame(sender,\ secret)$
                $\lor\ AddToPot(sender,\ money)$
                $\lor\ RemoveFromPot(sender,\ money)$
                $\lor\ (\exists\, g \in \text{BOOLEAN}\ \ : PlaceBet(sender,\ money,\ g))$
```

```
                 $\lor DecideBetWin(sender, secret2)$
                 $\lor DecideBetLoose(sender, secret2)$

$Spec \triangleq \forall op \in Int :$
          $\forall secret \in Int :$
             $\land Init(op)$
             $\land \Box[Step(secret)]_{\langle\rangle}$
```